IC Compiler Laboratory Exercise

Design Preparation

```
unix% tar -zxvf icc_lab_T90.gz
```

Table I Lab1

DESIGN DATA	FILE OR DIRECTORY
Gate Level Netlist	~/icc_lab/Lab1/design_data/CHIP_syn.v
IO Constraint File	~/icc_lab/Lab1/design_data/io.tdf
Timing Constraint File	~/icc_lab/Lab1/design_data/CHIP.sdc
Scandef Fille	~/icc_lab/Lab1/design_data/CHIP.scandef
Technology File (CIC Design Kit)	**/cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2/CIC/ICC/tsmc090_9lm_2thick_cic.tf**
Layer Mapping File	**/cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2/CIC/ICC/macro.map**
Antenna Rules	**/cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2/CIC/ICC/antenna_9lm_CIC.clf**
Reference Library (Memory)	~/icc_lab/Lab1/ref_lib/nco_table_cos ~/icc_lab/Lab1/ref_lib/nco_table_log
Reference Library (Core)	**/cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2/CIC/ICC/tsmc090nvt_fram** **/cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2/CIC/ICC/tsmc090hvt_fram**
Reference Library (IO)	**/cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2/CIC/ICC/tpzn90gv3** **/cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2/CIC/ICC/tpbn90v**
***TLU+ Data**	**/cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2/CIC/ICC/tluplus**
Script Files	~/icc_lab/Lab1/scripts/
Directory for Physical Verification	~/icc_lab/Lab1/verify/

Lab1-1 Design Setup

1. Change directory to ~/icc_lab/Lab1/run and invoke IC Compiler

   unix% cd ~/icc_lab/Lab1/run

   unix% Ricc_shell –gui

   (於 EDA Cloud 環境執行 EDA Tools 需加大寫 R，並且不能加&背景執行)

2. **“ File > Create Library ”**

New Library Name	CHIP
Technology File	/cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2 /CIC/ICC/tsmc090_9lm_2thick_cic.tf
Bus naming style	[%d]
Open library	enable

在Input reference libraries項目點選視窗 Add

在Select Directory視窗選擇

/cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2/CIC/ICC/tpbn90v 按 OK

/cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2/CIC/ICC/tpzn90gv3 按 OK

/cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2/CIC/ICC/tsmc090hvt_fram 按 OK

/cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2/CIC/ICC/tsmc090nvt_fram 按 OK

../ref_lib /nco_table_cos 按 OK

../ref_lib /nco_table_log 按 OK 選擇完成。

按 OK 關閉視窗。

3. **“ File > Import Designs ”**

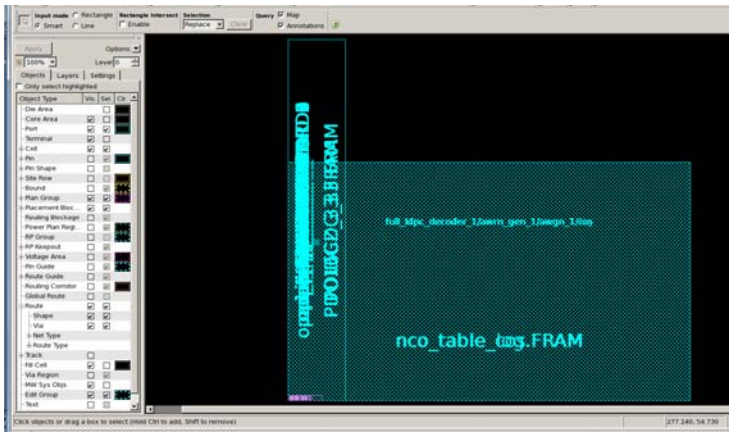
Import format	verilog
Import verilog files	../design_data/CHIP_syn.v
Top design name	CHIP

點選視窗 Add

在Select File視窗選擇 ../design_data/CHIP_syn.v 按 Open 選擇。

按OK。

可以看到ICC將整個design的CEL View建立出來。



4. **“ File > Set TLU+… ”**

Max TLU+ file	`/cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2/CIC/ICC/tluplus/crn90lp_1p09m+alrdl_mim_typical.tluplus`
Layer name mapping file between technology library and ITF file	`/cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2/CIC/ICC/tluplus/T90.map`

5. **“ File > Import >Read SDC…”**

Input file name	`../design_data/CHIP.sdc`
Version	`Latest`
Other	`default value`

6. **“ File > Save Design ”**

點選Save All

“ File > Save Design ”

點選Show advanced options

Save As	`Enable`
Save As Name	`design_setup`

按OK。

7. **“ File > Close Design ”**

“ File > Close Library ”

※若 lab1_1 無法完成，可在 Message/Input Area 輸入 ***source “../scripts/01_design_setup.tcl”***
以利後面的 lab 進行。

p.s. 執行前請先將 CHIP 資料夾刪除(在 Terminal 利用 rm –rf * 指令)

Lab1-2 Design Planning

1. Invoke IC Compiler (@path "~/icc_lab/Lab1/run ")

   unix% Ricc_shell –gui

   (於 EDA Cloud 環境執行 EDA Tools 需加大寫 R，並且不能加&背景執行)

2. **" File > Open Library"**

Library Name	CHIP
open library as read-only	disable
open reference library for writing	disable

   按 OK。

   " File > Open Design"

   選擇 CHIP

   按 OK。

3. 我們要加入 IO 及 Core 的 P/G pad, POC pad 及 Corner pad

   在 Message/Input Area 輸入

   ```
   source ../scripts/create_phy_cell.tcl
   ```

   按 enter。

4. Read TDF file，先切換至 LayoutWindows

   " Floorplan > Read Pin/Pad Physical Constraints "

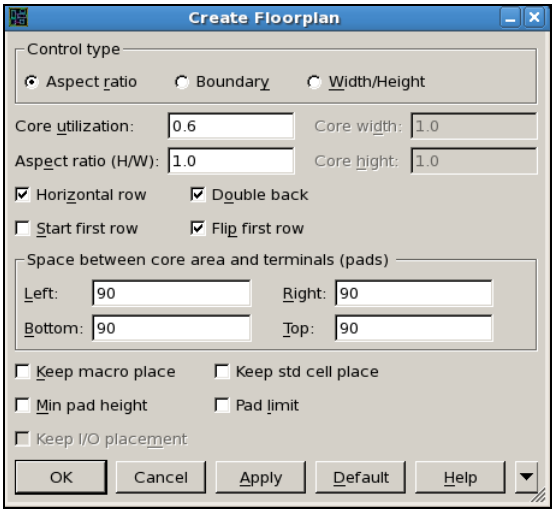
Input file name	../design_data/io.tdf

   按 OK。

   " Floorplan > Create Floorplan… "

Control type	Aspect ratio
Core utilization	0.6
Aspect ratio (H/W)	1
Horizontal row	Enable
Double back	Enable
Start first row	Disable
Flip first row	Enable
Core to left	90
Core to bottom	90
Core to right	90
Core to top	90

   按 OK。

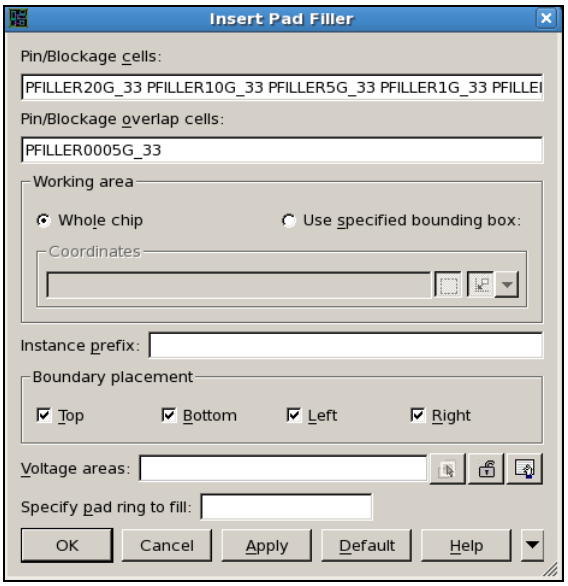


5. Insert pad filler

“ Finishing > Insert Pad Filler ”

Pin/Blockage cells	`PFILLER20G_33 PFILLER10G_33 \ PFILLER5G_33 PFILLER1G_33 \ PFILLER05G_33 PFILLER0005G_33` (由大至小排序)
Pin/Blockage overlap cells	`PFILLER0005G_33`
Boundary placement (Top)	`Enable`
Boundary placement (Bottom)	`Enable`
Boundary placement (Left)	`Enable`
Boundary placement (Right)	`Enable`
Other	`Default value`

按 OK。



6. **“ File > Save Design ”**

點選 Save All

“ File > Save Design ”

點選Show advanced options

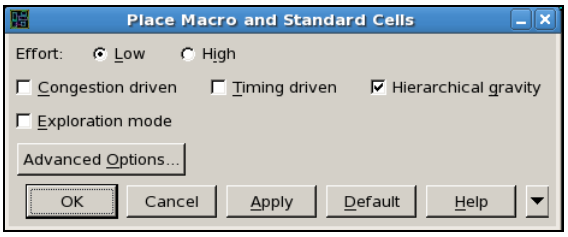
Save As	Enable
Save As Name	die_init

按OK。

7. **“ Placement > Place Macro and Standard Cells ” (如找不到選項請設定 File > Task > All Task)**

Effort	Low
Congestion driven	disable
Timing driven	disable
Hierarchical gravity	enable

按 OK。

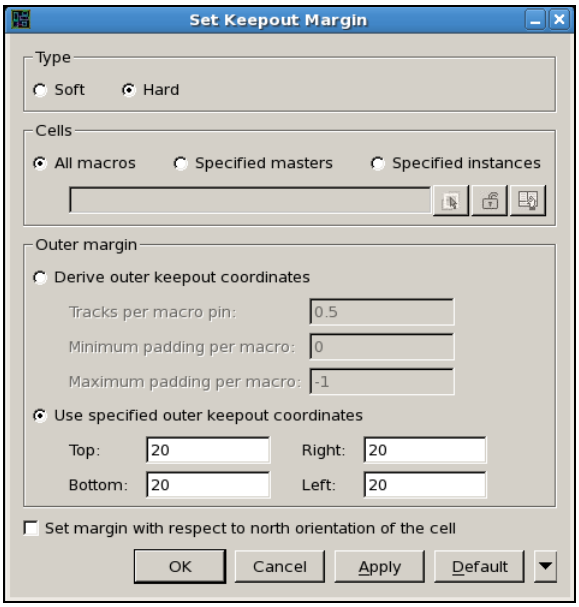


可以觀察到所有的 Macro 與 Standard Cell 皆放置完成

先設定所有的 macro 四周圍各 20um 的 keepout margin

“ Placement > Set Keepout Margin... ”

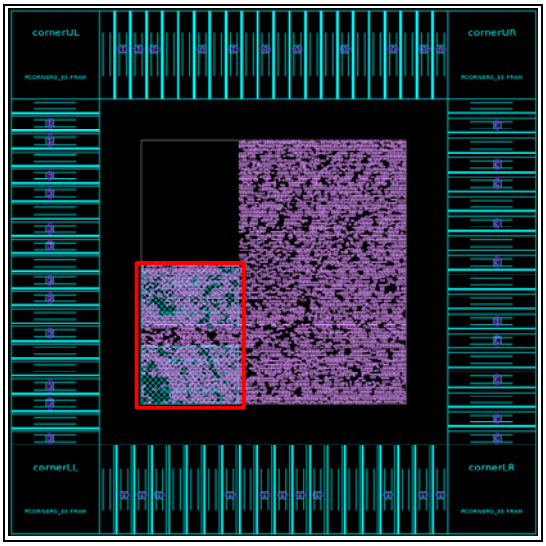
Type	Hard
Cell	All macros
Use specified outer keepout coordinates	Top: 20 Right: 20 Bottom: 20 Left: 20



為了方便後續步驟，我們將 Macro 統一放置在固定位置

在 Message/Input Area 輸入

```
move_objects -x 280 -y 446 [get_cells -hier log]
move_objects -x 280 -y 280 [get_cells -hier cos]
```



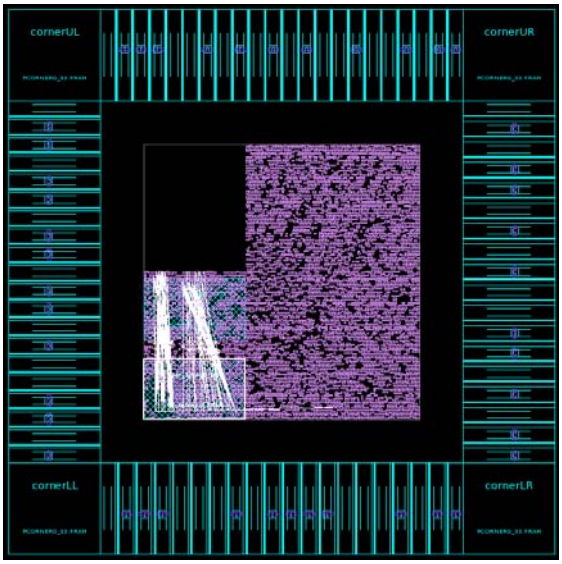
擺放完成後將所有 macro fix 住

在 Message/Input Area 輸入

```
set_dont_touch_placement [all_macro_cells]
```

點選 Flylines 

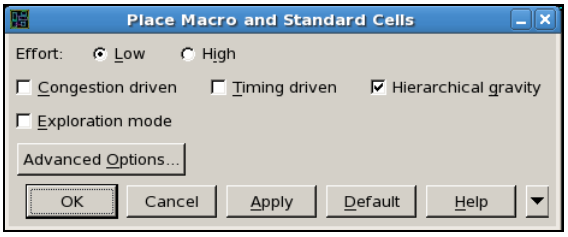
接著點選 Macro，觀察 Macro 連線狀態，觀察完關閉。



8. 有些 cell overlap 在一起，執行 Incremental placement 將 memory 底下的 cell 擺開

“ Placement > Place Macro and Standard Cells ”

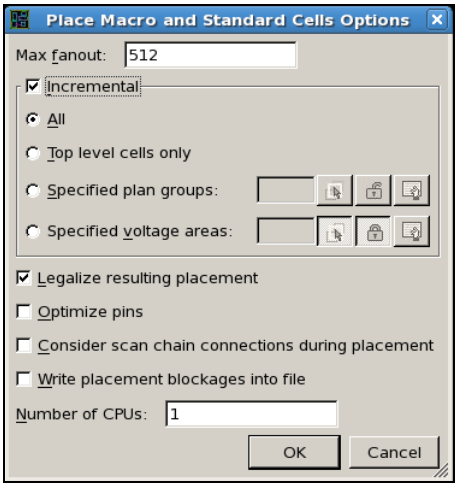
Effort	Low
Congestion driven	disable
Timing driven	disable
Hierarchical gravity	enable



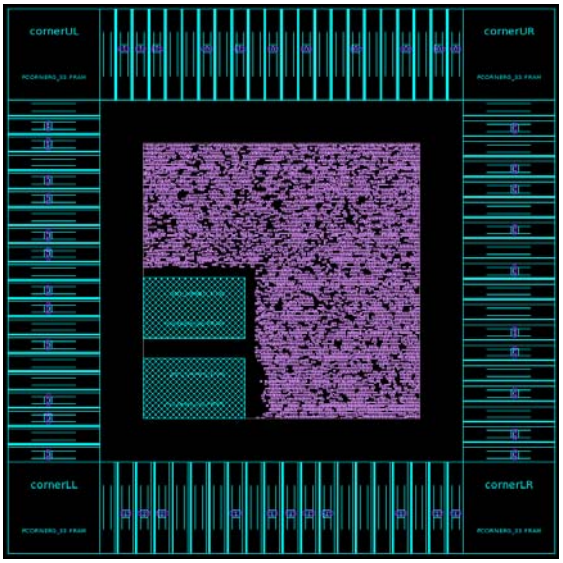
選取 Advanced Options Tab

Incremental	enable
Other	Default Value

按 OK。



可觀察到 memory 底下的 cell 已經擺開。



9. **“ File > Save Design ”**

點選 Save All

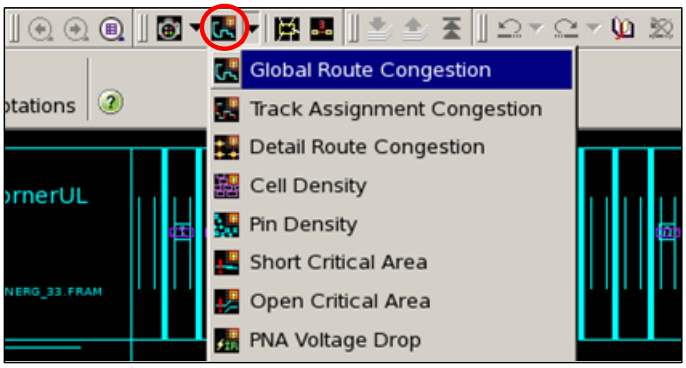
“ File > Save Design ”

點選Show advanced options

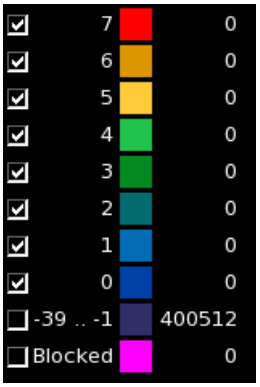
Save As	Enable
Save As Name	Before_PNS

按OK。

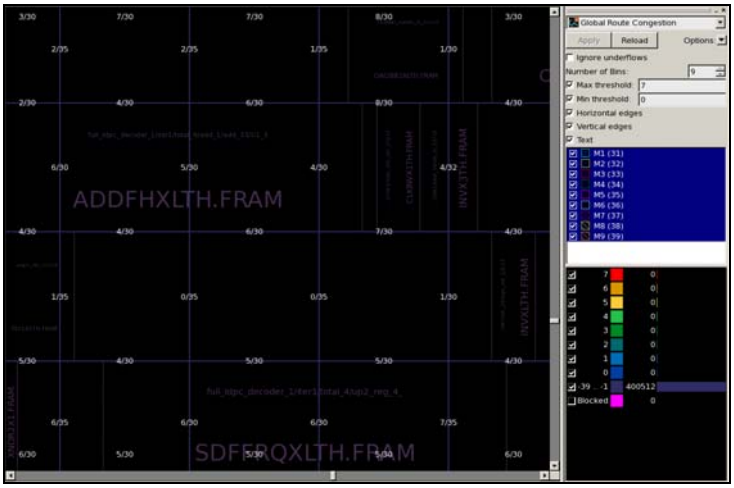
10. 分析 Congestion 狀態，點選 Global Route Congestion 視窗，再按下 reload，觀察完關閉



由於預設只有顯示發生 Congestion 的地方，如果觀察不到東西，可以檢查是否因為沒有 Congestion 發生。



如果要觀察沒有發生 Route Congestion 的地方，可將沒有發生 Congestion(意即 Congestion 為負) 部分打勾，接著按 Apply，之後再 Zoom in 畫面即可觀察到，觀察完關閉。



Note: 如果有 congestion 問題，如發生在 macro 附近，可以試著將 keepout margin 加大，或是將 macro 與 macro 之間的距離加大。如發生在 standard cell，可等至 placement 解決。

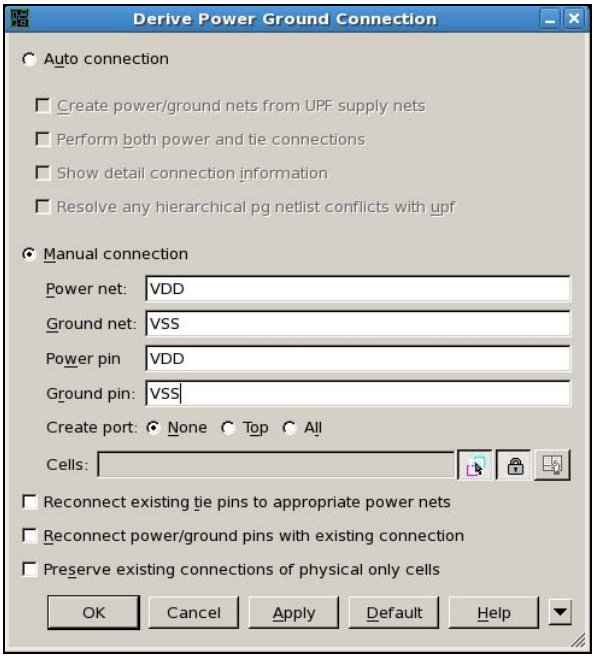
11. 連接 Cell 的 P/G nets

“ Preroute > Derive PG Connection ”

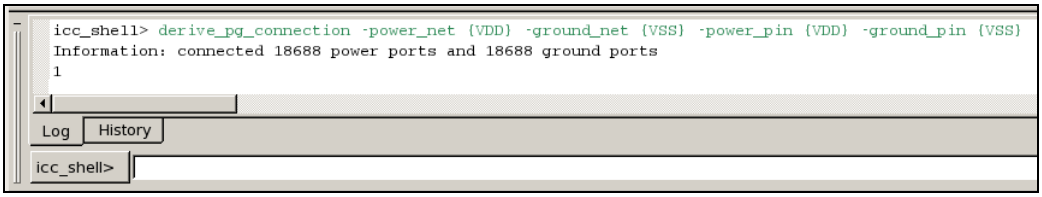
Manual connection	`selected`
Power net	VDD
Ground net	VSS
Power pin	VDD
Ground pin	VSS
Other	`Default value`

按 Apply

按OK。



可觀察 IC Compiler 的訊息視窗，顯示執行 Power / Ground Connection 後的訊息



12. 設定 PNS Constraints

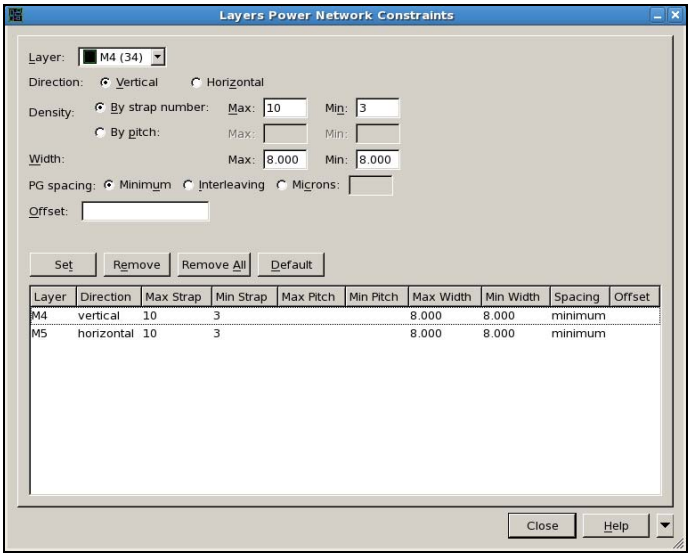
“ Preroute > Power Network Constraints > Strap Layers Constraints”

layer	**M5**
Direction	`Horizontal`
Density (By strap number)	`Max:10 Min:3`
Width	`Max:8 Min:8`

按 Set 再選

layer	**M4**
Direction	Vertical
Density (By strap number)	Max:10 Min:3
Width	Max:8 Min:8

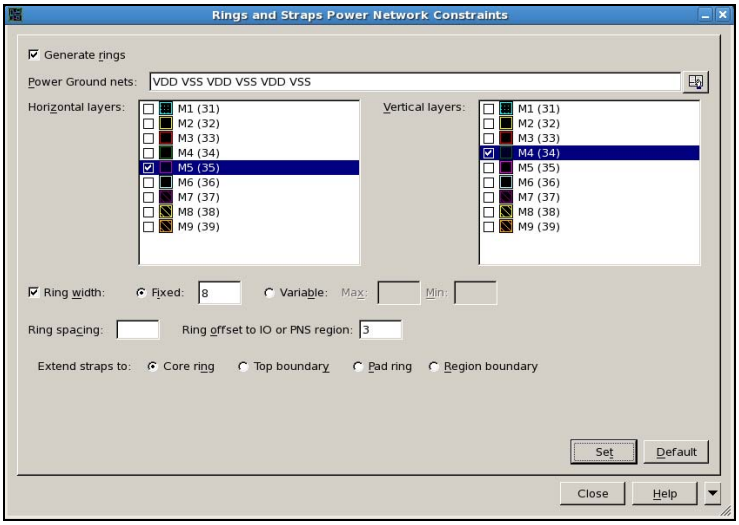
按 Set 後按 Close 離開。



“ Preroute > Power Network Constraints > Ring Constraints”

Power Ground nets	VDD VSS VDD VSS VDD VSS (6 圈共 3 對)
Horizontal layers	M5
Vertical layers	M4
Ring width	enable
Variable	Fixed (8 um)
Ring offset to IO or PNS region	3
Variable	Core ring

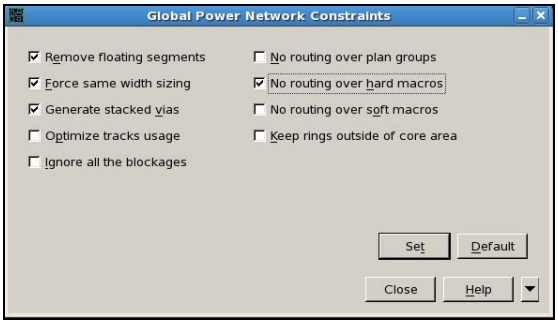
按 Set 後按 Close 離開。



“ Preroute > Power Network Constraints > Global Constraints”

No routing over hard macros	enable
Other	Default value

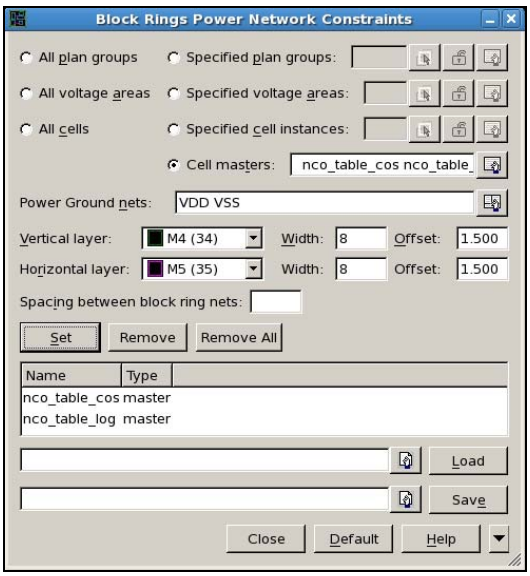
按 Set 後按 Close 離開。



“ Preroute > Power Network Constraints > Block Ring Constraints”

Cell master	selected (nco_table_cos nco_table_log)
Other	Default value
Power Ground nets	VDD VSS
Vertical layer	M4 (Width:8 Offset:1.5)
Horizontal layer	M5 (Width:8 Offset:1.5)

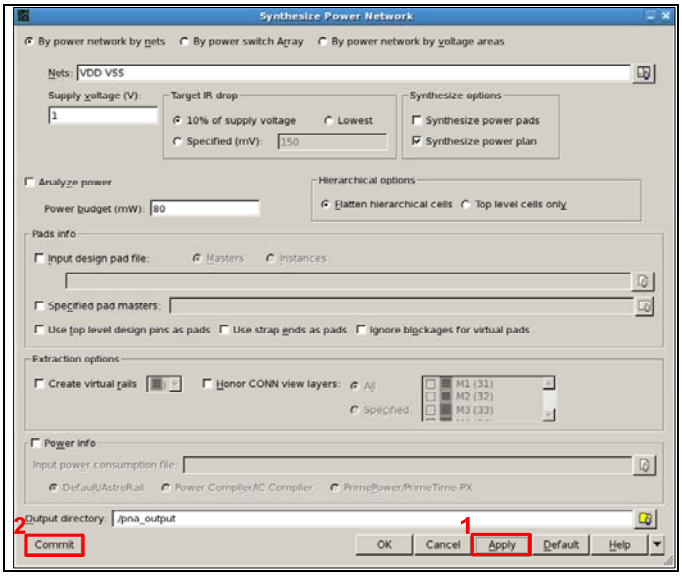
按 Set 後按 Close 離開。



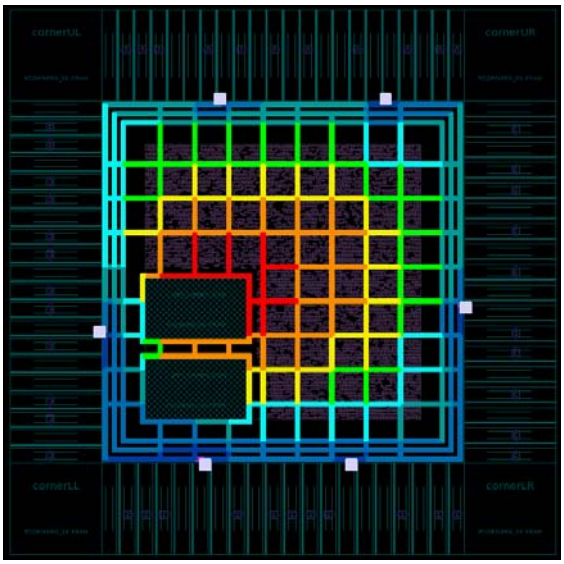
“ Preroute > Synthesize Power Network”

By power network by nets	VDD VSS
Supply voltage	1
Target IR drop	10% of supply voltage
Power budget (mW)	80
Other	Default value

按 Apply 觀察 IR drop map 後，按 Commit 連接 P/G net。

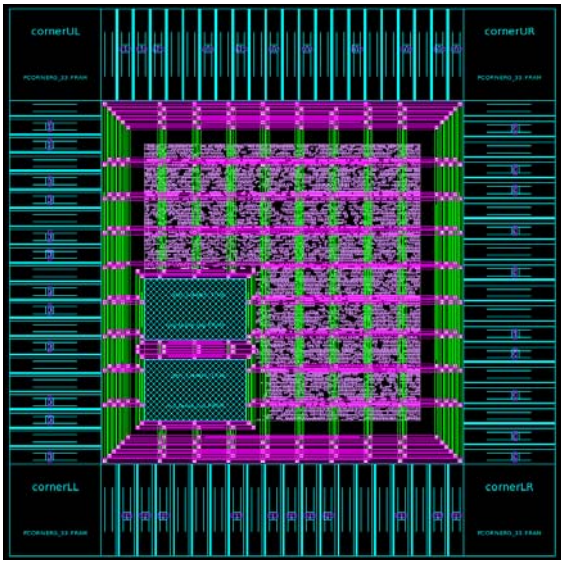


IR drop map:



Zoon In 畫面可觀察到 IR drop 的數值

Commit 後，可觀察到先前設定的 Power strap、Power ring 以及 Macro 的 Block ring 已經加上去。



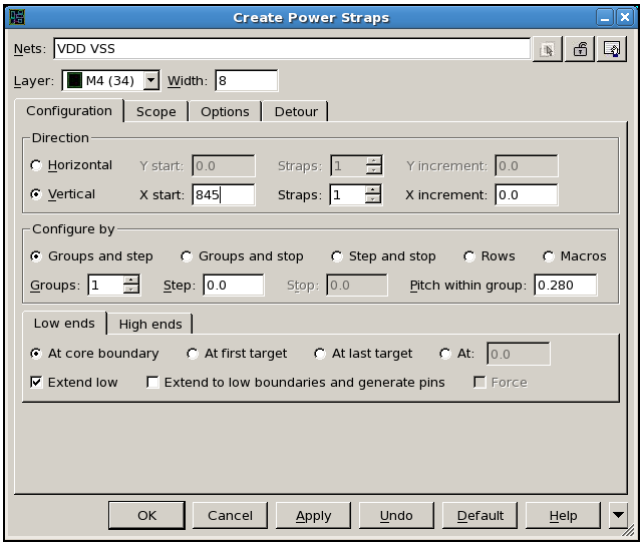
13. 手動建立 CHIP 右邊的 Strap 連接

“ Preroute > Create Power Straps…”

Nets	VDD VSS
Layer	M4
Width	8

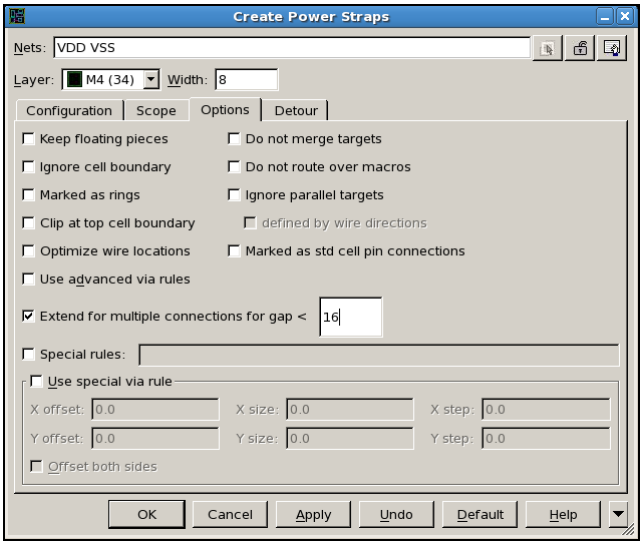
點選 Configuration

Direction	Vertical
X start	845



點選 Options

Extend for multiple connections for gap	Enable
Gap	16



按 Apply

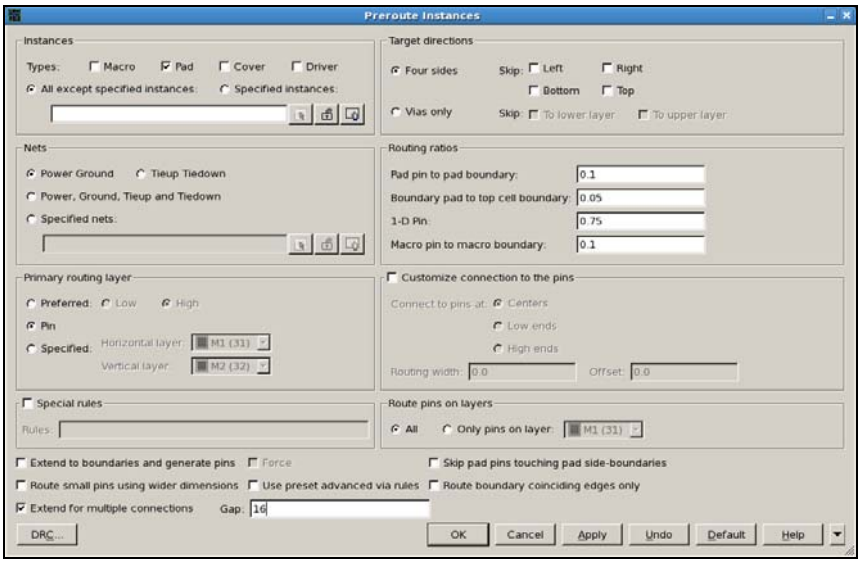
按 OK。

14. 建立 I/O Pad 到 P/G Ring

“ Preroute > Preroute Instances”

Instances	Pad**(只勾選 Pad 其餘 disable)**
Target directions	default value
Primary routing layer	**Pin**
Extend for multiple connections	Enable
Gap	16

按 Apply 之後再按 **Default**



建立 Block Ring 到 P/G Ring 的連接

“ Preroute > Preroute Instances”

Instances Types	Macro**(只留 Macro 其餘 disable)**
Primary routing layer	**Pin**

按 Apply

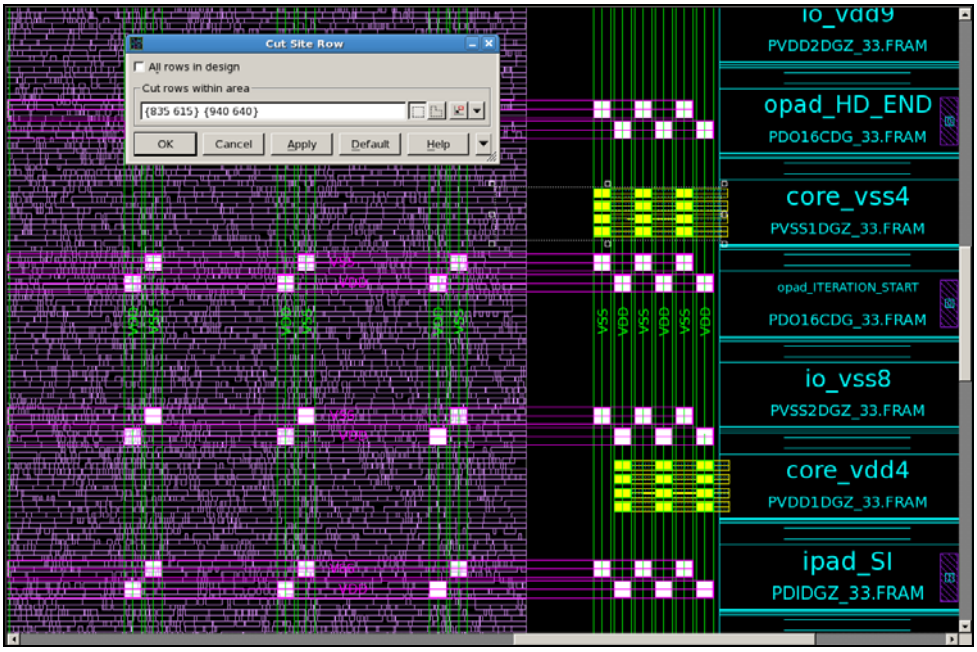
按 OK

15. 將 rail 與 I/O Pad 到 P/G Ring 之間的 row 刪除

“ Floorplan > Cut site Row”

All rows in design	**disable**
Coordinates	{835 615} {940 640}

按 Apply。

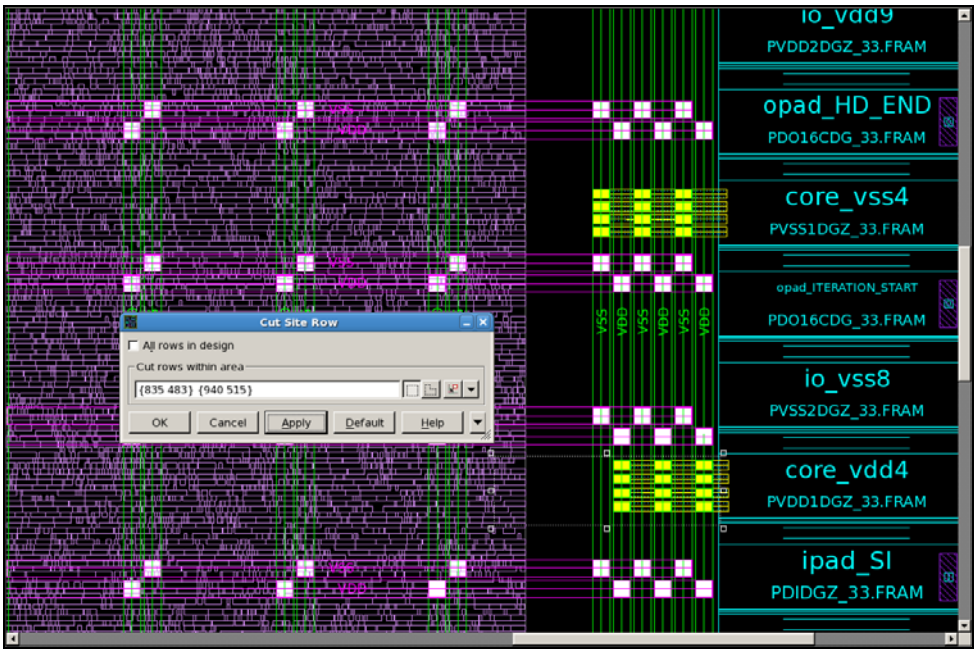


再修改

All rows in design	**disable**
Coordinates	{835 483} {940 515}

按 Apply。

按 OK



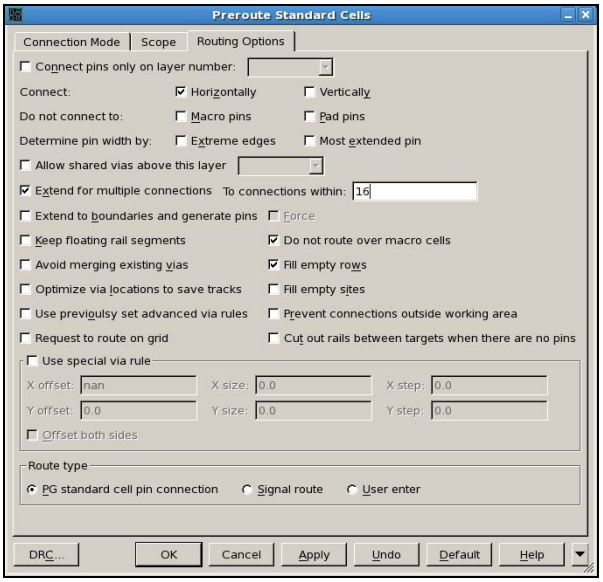
16. Preroute Standard Cell Rail

“ Preroute > Preroute Standard Cells…”

選 Routing Option

Extend for multiple connections	Enable
To connections within	16
Keep floating rail segments	disable
Do not route over macro cells	enable
Fill empty rows	enable

按 OK 離開。



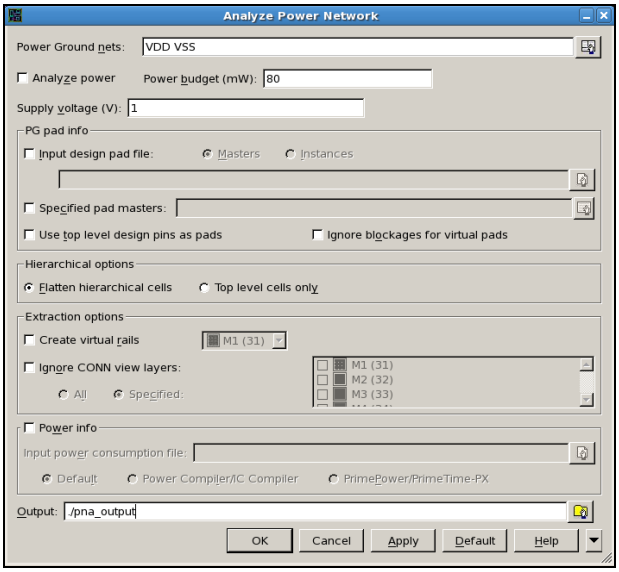
17. 分析做完整個 Power network synthesis 後的 IR drop 情形

“ Preroute > Analyze Power Network”

Power Ground nets	VDD VSS
Supply voltage	1
Power budget (mW)	80
Other	Default value

按 OK 觀察 IR drop map，觀察 IR drop 最嚴重地點為何處? IR drop 最嚴重的量多少?

觀察後關閉 PNA Voltage Drop 選單



18. 設定 Strap 下不要擺放 Standard Cell
    在 Message/Input Area 輸入

```
set_pnet_options -partial "M4 M5"
create_fp_placement -incremental all
```

19. **“ File > Save Design ”**
    點選 Save All

    “ File > Save Design ”
    點選Show advanced options

Save As	Enable
Save As Name	design_planning

    按OK。

20. **“ File > Close Design ”**
    “ File > Close Library ”

※若 lab1_2 無法完成，可在 Message/Input Area 輸入 ***source “../scripts/02_design_planning.tcl”***
以利後面的 lab 進行。

Lab1-3 Placement

1. Invoke IC Compiler

   unix% Ricc_shell –gui

   (於 EDA Cloud 環境執行 EDA Tools 需加大寫 R，並且不能加&背景執行)

2. **“ File > Open Library”**

Library Name	CHIP
open library as read-only	disable
open reference library for writing	disable

   按 OK。

   “ File > Open Design”

   選擇 CHIP

   按 OK。

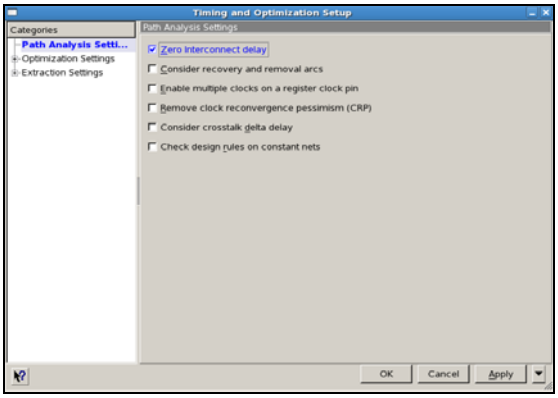
3. **“ File > Import >Read SDC…”**

Input file name	../design_data/CHIP.sdc
Version	Latest
Other	default value

4. 設定 zero interconnect delay 屬性為 true

   “Timing > Timing and Optimization Setup… ”

Zero Interconnect delay	enable
Other	default value



5. Report timing

   在 Message/Input Area 輸入

   `report_timing`

6. Check all constraint violations:

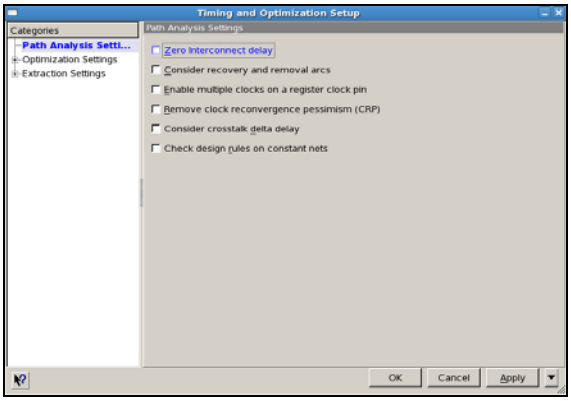
   在 Message/Input Area 輸入

   `report_constraint –all`

7. 設定 zero interconnect delay 屬性為 false

“Timing > Timing and Optimization Setup… ”

Zero Interconnect delay	`disable`
Other	`default value`

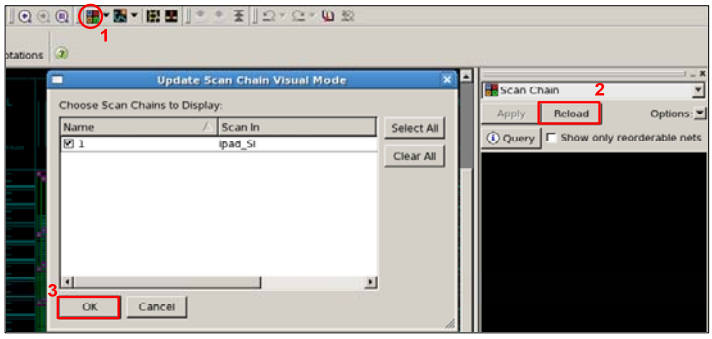


8. Load the SCANDEF file:

在 Message/Input Area 輸入

```
read_def ../design_data/CHIP.scandef
report_scan_chain
check_scan_chain
```

9. 點選 Scan Chain，在 Scan Chain 選單按 Reload 載入 Scan Chain 的資訊，接著在”Update Scan Chain Visual mode 視窗將 Choose Scan Chains to Display 將 Scan In 打勾，按 OK。在 LayoutWindows 觀察 scan chain，觀察完並關閉。



10. 產生 Power report:

在 Message/Input Area 輸入

```
report_power
```

Cell Internal Power	
Net Switching Power	
Total Dynamic Power	
Cell Leakage Power	

11. Setup threshold voltage group attribute and report it:
    在 Message/Input Area 輸入

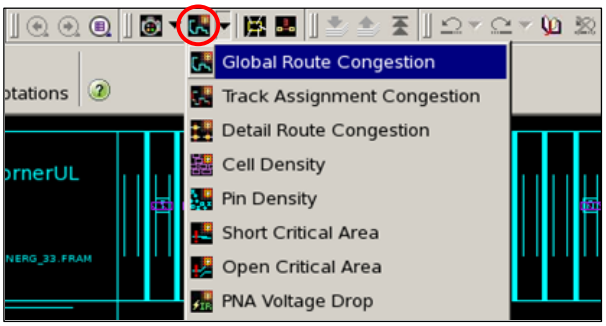
```
source ../scripts/group_MVT.tcl
```

Threshold voltage group	Number of Cells	Percentage (%)
HVt		
IO		
MACRO		
RVt		

12. Read the SAIF file of the design:
    在 Message/Input Area 輸入

```
read_saif -input ../design_data/CHIP.saif -instance_name test_top/CHIP_1
```

13. 分析 Congestion 狀態，點選 Global Route Congestion 視窗，並按 reload，觀察完關閉



14. The following commands should be executed before performing place_opt.
    在 Message/Input Area 輸入

```
check_physical_design –stage pre_place_opt
```

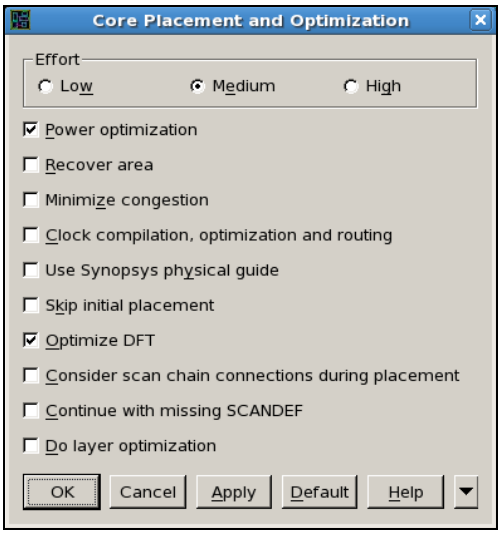
15. Setup the tie cell option:
    在 Message/Input Area 輸入

```
source ../scripts/add_tie.tcl
```

16. **“ Placement > Core Placement and Optimization… ”**

Effort	Medium
Power optimization	enable
Optimize DFT	enable

    按 OK。

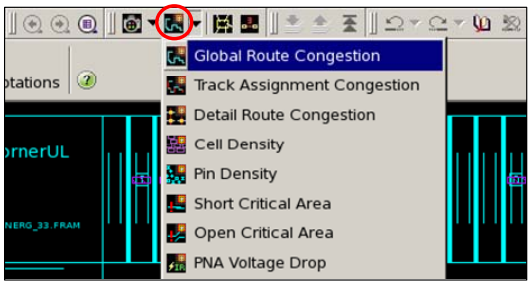


或者

在 Message/Input Area 輸入

```
place_opt -optimize_dft -power
```

17. 分析 Congestion 狀態，點選 Global Route Congestion 視窗，並按 reload，觀察完關閉



Note: 如果有 congestion 問題，請參考講義第 4 章後面 Refinement Flow

18. Report timing

在 Message/Input Area 輸入

```
report_timing
```

Note: 如果有 timing 問題請下 place_opt

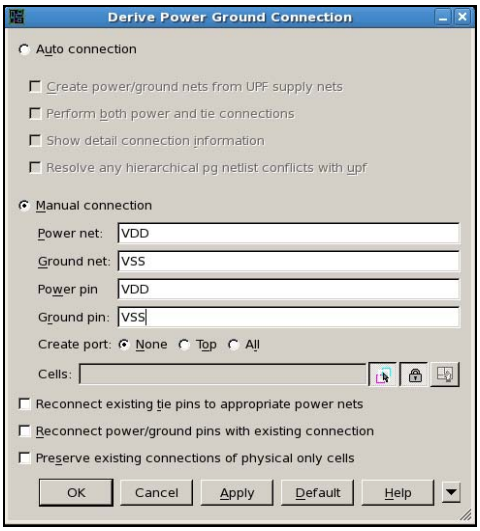
19. 連接 Cell 的 P/G nets

“ Preroute > Derive PG Connection ”

Manual connect	selected
Power net	VDD
Ground net	VSS
Power pin	VDD
Ground pin	VSS
Other	Default value

按 Apply。

按OK。



20. Report power and threshold voltage group:
在 Message/Input Area 輸入
`report_power`

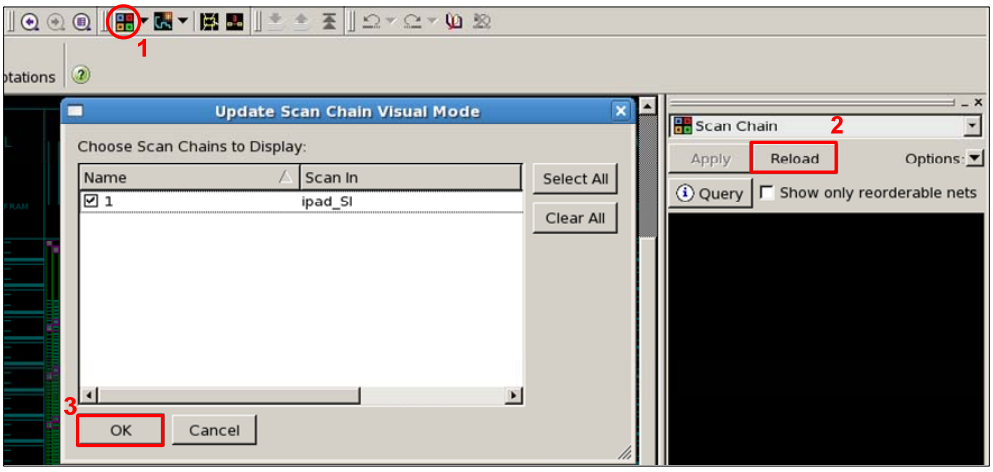
Cell Internal Power	
Net Switching Power	
Total Dynamic Power	
Cell Leakage Power	

`report_threshold_voltage_group`

Threshold voltage group	Number of Cells	Percentage (%)
HVt		
IO		
MACRO		
RVt		

比較步驟 31 與 32 的表格數據，造成兩者的差異原因為何?

21. 點選 Scan Chain，在 Scan Chain 選單按 Reload 載入 Scan Chain 的資訊，接著在”Update Scan Chain Visual mode 視窗將 Choose Scan Chains to Display 將 Scan In 打勾，按 OK。在 LayoutWindows 觀察 scan chain，觀察完並關閉。



22. **“ File > Save Design**

    點選 Save All

    “ File > Save Design ”

    點選Show advanced options

Save As	Enable
Save As Name	placement

    按 OK。

23. **“ File > Close Design ”**

    “ File > Close Library ”

    ※若 lab1_3 無法完成，可在 Message/Input Area 輸入 ***source “../scripts/03_placement.tcl”*** 以利後面的 lab 進行。

Lab1-4 CTS

1. Invoke IC Compiler

   unix% Ricc_shell –gui

   (於 EDA Cloud 環境執行 EDA Tools 需加大寫 R，並且不能加&背景執行)

2. **“ File > Open Library”**

Library Name	CHIP
open library as read-only	disable
open reference library for writing	disable

   按 OK。

   “ File > Open Design”

   選擇 CHIP

   按 OK。

3. 由於 design 中含有除頻器，設定 clock pin 為 hierarchical pin.

   在 Message/Input Area 輸入

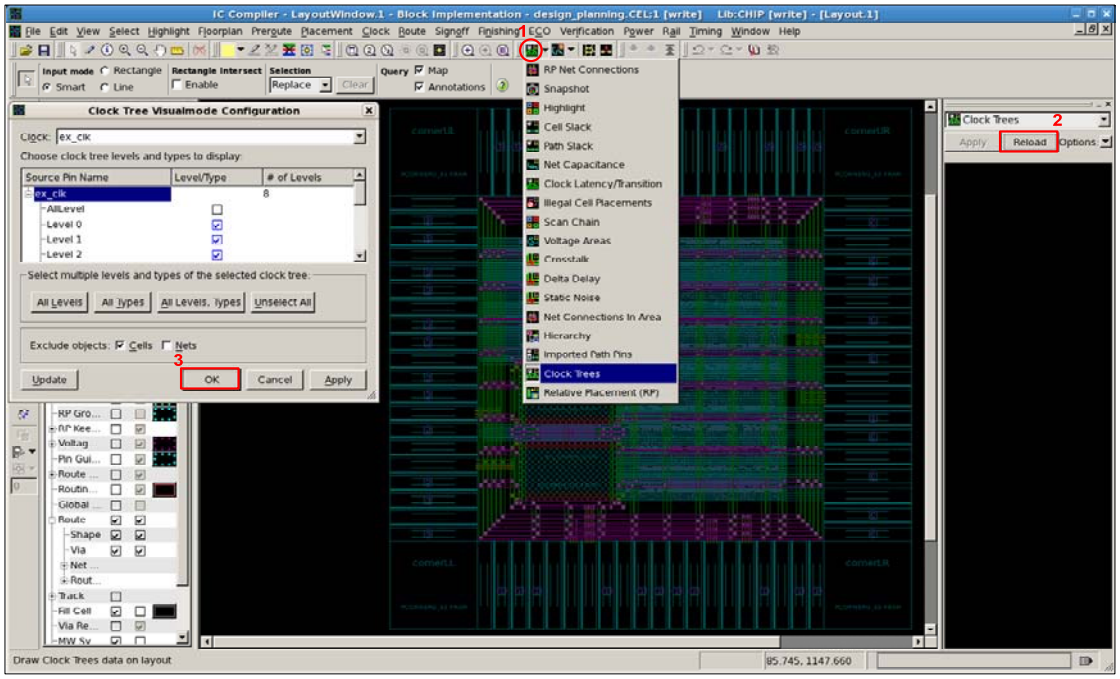
   ```
   set cts_enable_clock_at_hierarchical_pin true
   ```

4. 設定 Clock gating cell

   在 Message/Input Area 輸入

   ```
   identify_clock_gating
   report_clock_gating
   ```

5. 點選 Clock Trees，在 Clock Trees 選單按 Reload 載入 Clock Trees 的資訊，接著在”Clock Tree Visualmode Configuration 按 OK。在 LayoutWindows 觀察 Clock Tree，觀察完並關閉。



6. The following commands should be executed before performing CTS.
   在 Message/Input Area 輸入
   `check_physical_design –stage pre_clock_opt`
   檢查是否有 error?

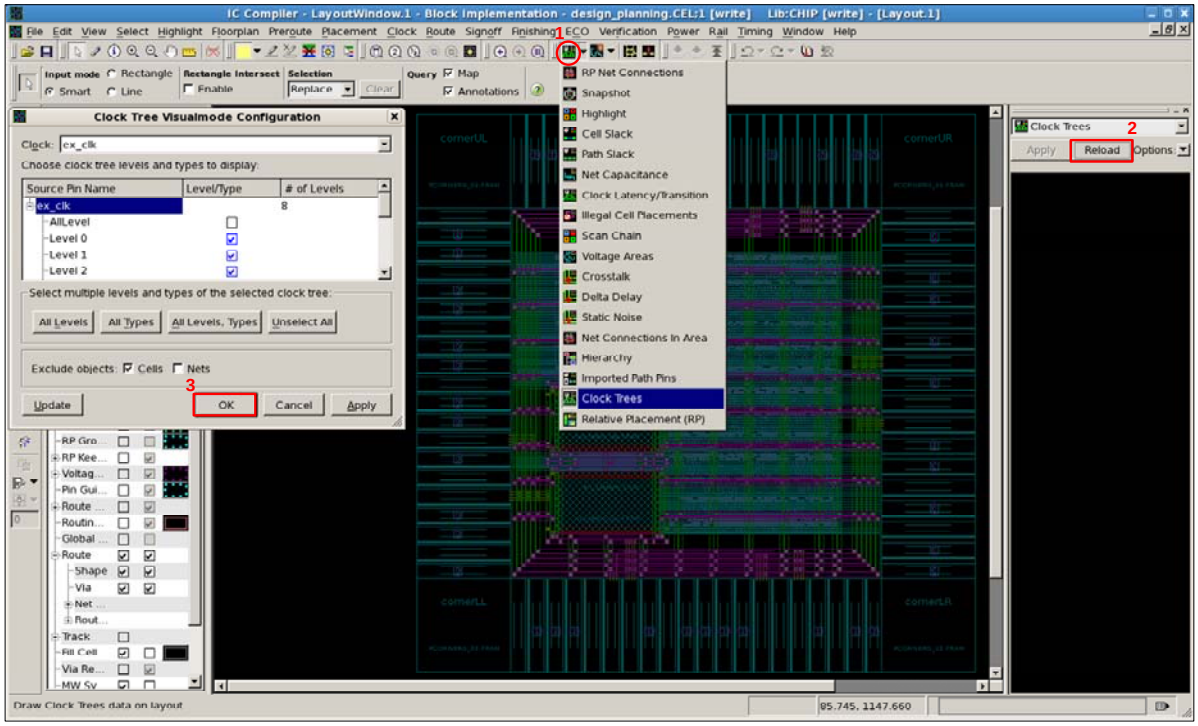
   `report_constraint -all`
   觀察是否有 violation?

7. Turn on hold time fixing.
   在 Message/Input Area 輸入
   `set_fix_hold [all_clocks]`

8. Report timing
   在 Message/Input Area 輸入
   `report_timing`

9. Perform CTS optmization:
   在 Message/Input Area 輸入
   `clock_opt -fix_hold_all_clocks  -no_clock_route`

10. 點選 Clock Trees，在 Clock Trees 選單按 Reload 載入 Clock Trees 的資訊，接著在"Clock Tree Visualmode Configuration 按 OK。在 LayoutWindows 觀察 Clock Tree，觀察完並關閉。



11. Report timing

在 Message/Input Area 輸入

```
report_timing
```

12. Review the global skew after CTS.

```
report_clock_tree  -summary
```

	Skew	Longestpath
ex_clk		
in_clk		
int_clk		

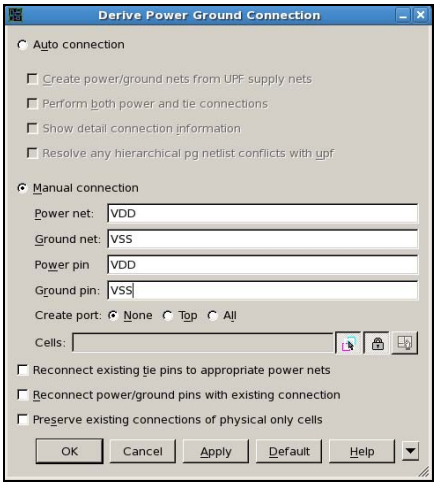
13. 連接 Cell 的 P/G nets

“ Preroute > Derive PG Connection ”

Manual connect	`selected`
Power net	VDD
Ground net	VSS
Power pin	VDD
Ground pin	VSS
Other	`Default value`

按 Apply。

按OK。



14. **“ File > Save Design**

點選 Save All

“ File > Save Design ”

點選Show advanced options

Save As	`Enable`
Save As Name	`cts`

按 OK。

15. **“ File > Close Design ”**
“ File > Close Library ”

※若 lab1_4 無法完成，可在 Message/Input Area 輸入 ***source “../scripts/04_cts.tcl”*** 以利後面的 lab 進行。

Lab1-5 Route

1. Invoke IC Compiler

   unix% Ricc_shell –gui

   (於 EDA Cloud 環境執行 EDA Tools 需加大寫 R，並且不能加&背景執行)

2. **“ File > Open Library”**

Library Name	CHIP
open library as read-only	disable
open reference library for writing	disable

   按 OK。

   “ File > Open Design”

   選擇 CHIP

   按 OK。

3. Analyze the design for timing (setup and hold)

   在 Message/Input Area 輸入

   `report_timing`

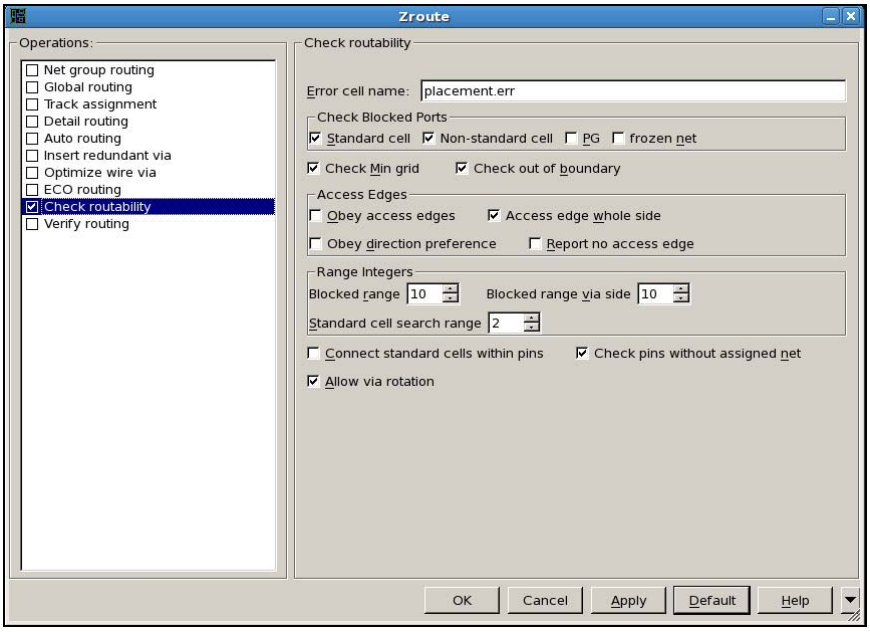
   觀察 Timing 有無 violation?

4. Check the routeability of the design

   “ Route > Check Routability…”

   按 OK。

   觀察是否有 error?



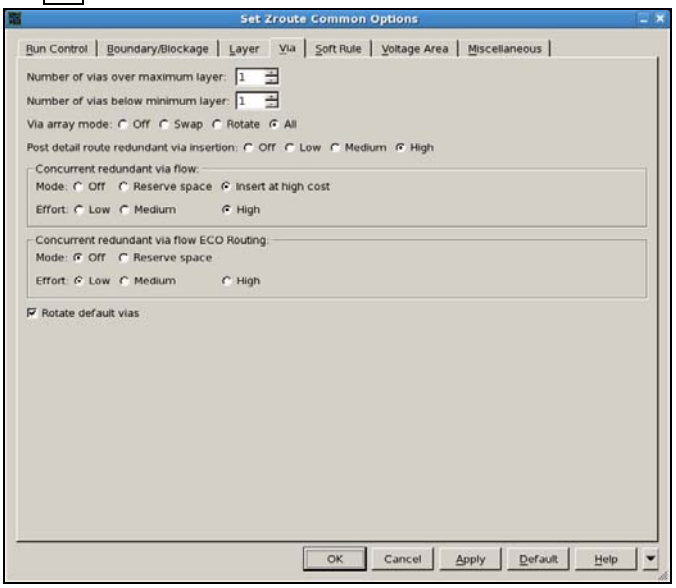
5. 設定在 routing 時的選項，並設定在 routing 時就加入 redundant via

“ Route > Routing Setup > Set common Route Option

選 Via Tab

Post detail route redundant via insertion	`High`
Concurrent redundant via flow	`Mode: Insert at High Cost`
Concurrent redundant via flow	`Effort: High`

按 OK。



6. To install a set of antenna rules, source the following command file

在 Message/Input Area 輸入

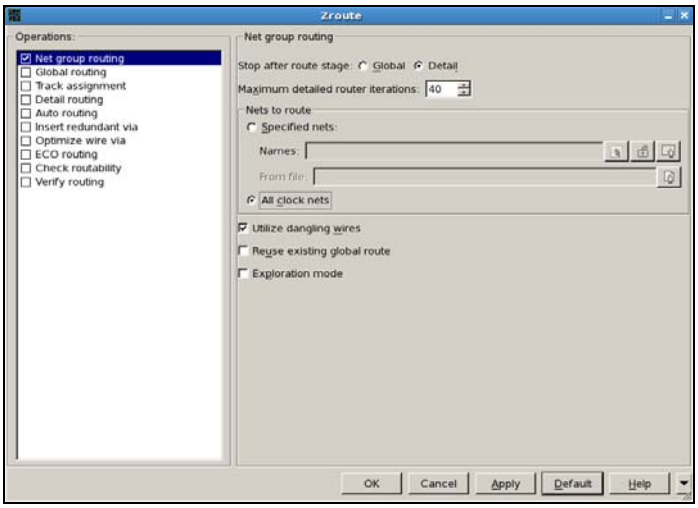
```
source /cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2/CIC/ICC/antenna_9lm_CIC.clf
```

7. Perform clock net routing

“Route > Net Group Route…“

Net to route	`All clock nets`
other	`default value`

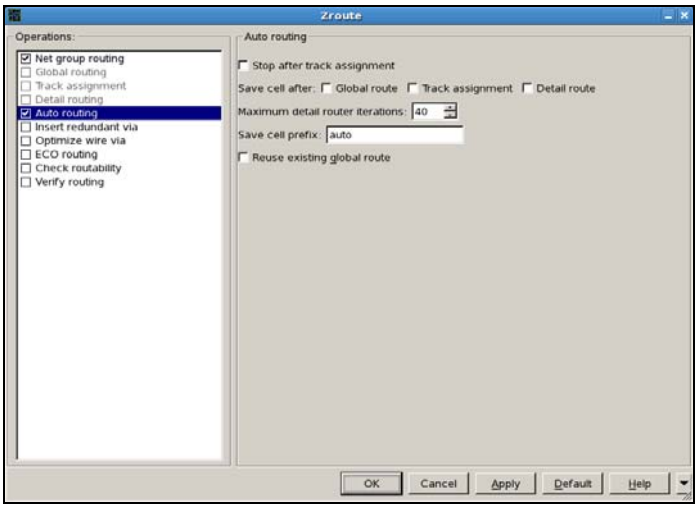
按 OK。



8. Perform auto routing

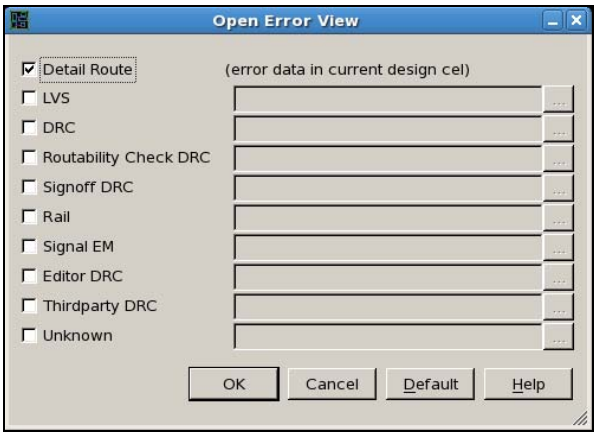
“Route > Auto Route…“

按 OK。



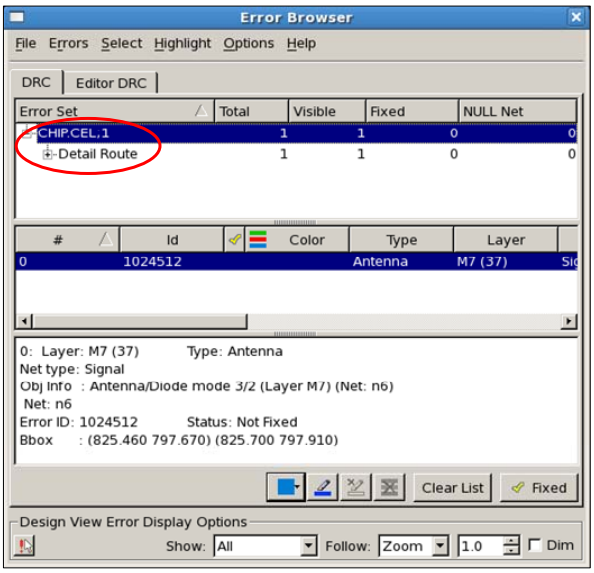
9. DRC violation error checking and fixing

“Verification > Error Browser…”



按 OK。

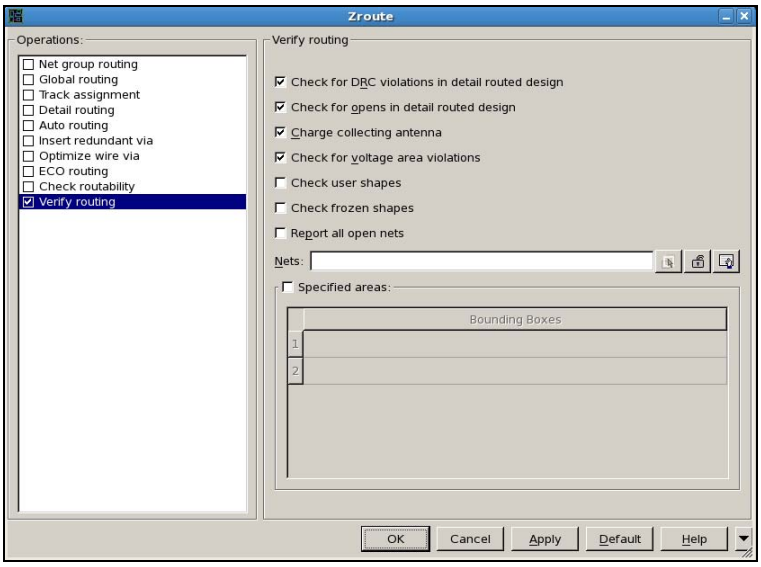
點取有 DRC error 的地方，會在 layout 中標示出來。



10. Recalculate the number of DRC violations as seen by the router

“Route > Verify Route…”

按 OK。



執行後如還有 DRC Violation，訊息如下

Verify Summary:

Total number of nets = 17204, of which 0 are not extracted

Total number of open nets = 0, of which 0 are frozen

Total number of excluded ports = 0 ports of 0 unplaced cells connected to 0 nets

0 ports without pins of 0 cells connected to 0 nets

0 ports of 0 cover cells connected to 0 non-pg nets

Total number of DRCs = 0

Total number of antenna violations = 1

Total number of voltage-area violations = no voltage-areas defined

Total number of tie to rail violations = not checked

Total number of tie to rail directly violations = not checked

如有 **Open net** 或 **DRC Violation** 則須執行 **Step 11**，如沒有 **DRC Violation**，則可跳至 **Step 12**

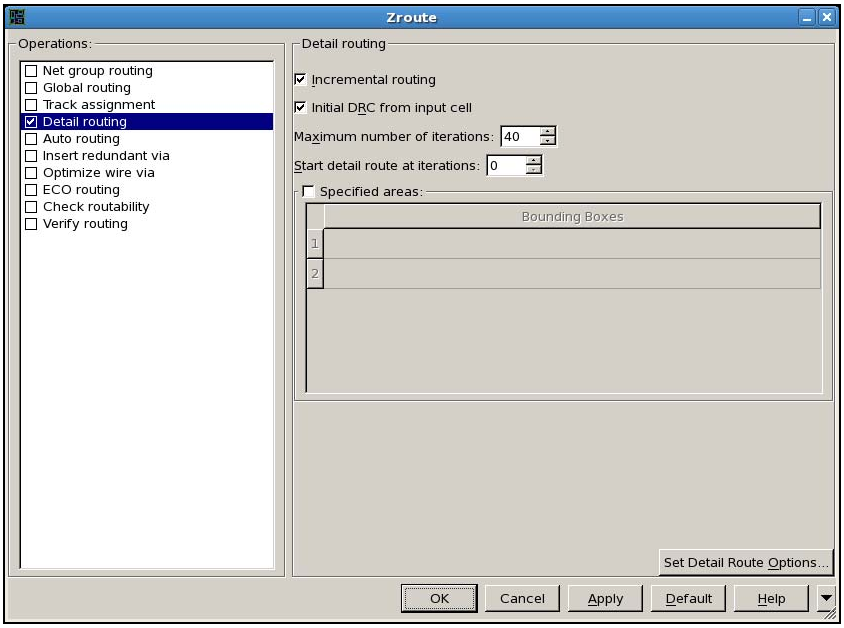
如有 **Antenna violation** 則在 **DFM** 階段，加入 **Antenna diode** 解決

11. Run a search & repair operation

“Route > Detail Route”

Incremental routing	enable
Initial DRC from input cell	enable

按 OK。



注意!! 執行 Detail Route 來修正 DRC violation 時，當 Initial DRC from input cell 選項為 enable 時，請務必確認有先做過 verify route

如有 DRC violation，可使用 Detail Route 並下 Incremental routing 與 Initial DRC from input cell 選項來修正

12. Analyze the design for timing and fixing

    在 Message/Input Area 輸入

    `report_timing`

    如有 Timing Violation 則請使用下列方式來修正

    “Route > Core Routing and Optimization…” 按 OK。(即指 **route_opt** 或 **route_opt -incremental**)

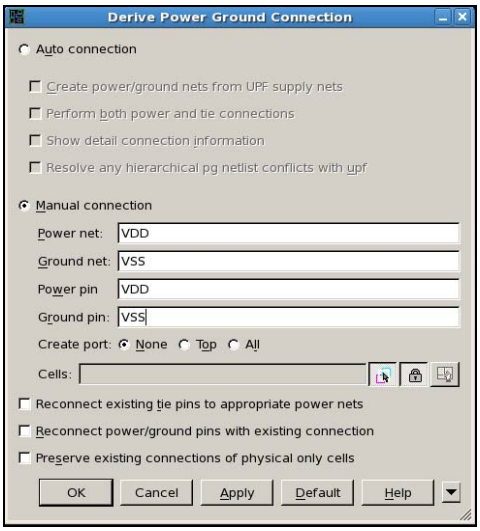
13. 連接 Cell 的 P/G nets

    “ Preroute > Derive PG Connection ”

Manual connect	`selected`
Power net	VDD
Ground net	VSS
Power pin	VDD
Ground pin	VSS
Other	`Default value`

    按 Apply。

    按OK。



14. **“File > Save Design**

    點選 Save All

    “ File > Save Design ”

    點選Show advanced options

Save As	Enable
Save As Name	route

    按 OK。

15. **“ File > Close Design ”**

    “ File > Close Library ”

※若 lab1_5 無法完成，可在 Message/Input Area 輸入 ***source “../scripts/05_route.tcl”*** 以利後面的 lab 進行。

Lab1-6 DFM

1. Invoke IC Compiler

   unix% Ricc_shell –gui

   (於 EDA Cloud 環境執行 EDA Tools 需加大寫 R，並且不能加&背景執行)

2. **“ File > Open Library”**

Library Name	CHIP
open library as read-only	disable
open reference library for writing	disable

   按 OK。

   “ File > Open Design”

   選擇 CHIP

   按 OK。

3. Insert standard cell filler

   在 Message/Input Area 輸入

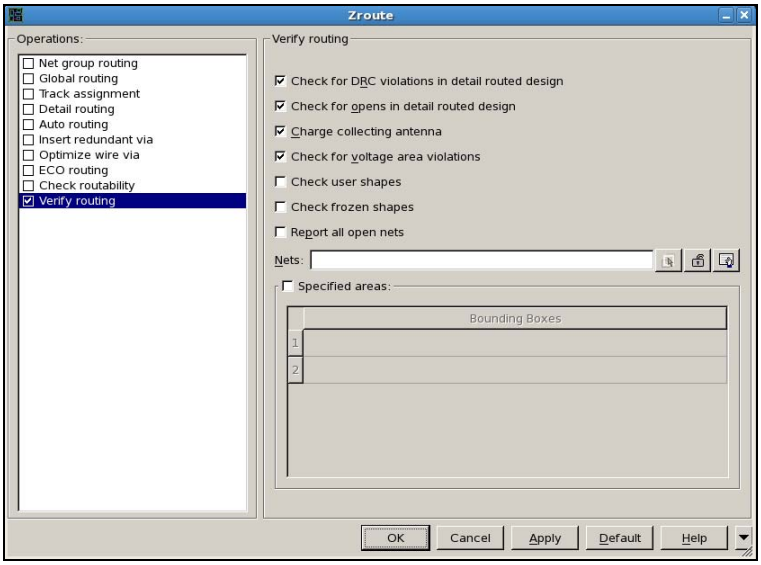
   `source ../scripts/addCoreFiller_MVT.cmd`

   (請執行兩次以上)

4. Verify insert standard cell filler 後的 DRC violation

   “Route > Verify Route…”

   按 OK。

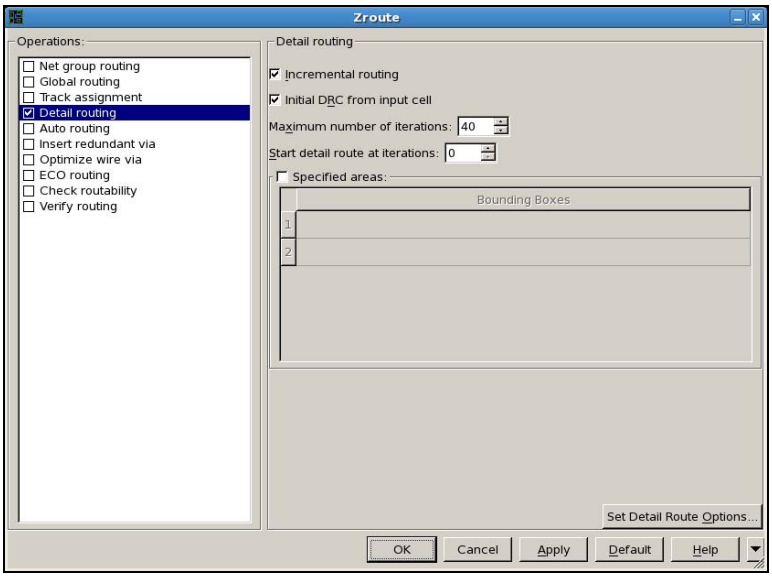


5. 如果有 Open net 或 DRC violation 則執行 DRC Violation fixing

“Route > Detail Route”

Incremental routing	`enable`
Initial DRC from input cell	`enable`

按 OK。



注意!! 執行 Detail Route 來修正 DRC violation 時，當 Initial DRC from input cell 選項為 enable 時，請務必確認有先做過 verify route

6. Analyze the design for timing and fixing

在 Message/Input Area 輸入

```
report_timing
```

但如果有 timing violation 時，則請使用下列方式來修正

“Route > Core routing and Optimization…” 按 OK。(即指 **route_opt** or **route_opt -incremental**)

7. 如有 Antenna Violation 則需插入 antenna diode，沒有即可跳過此步驟

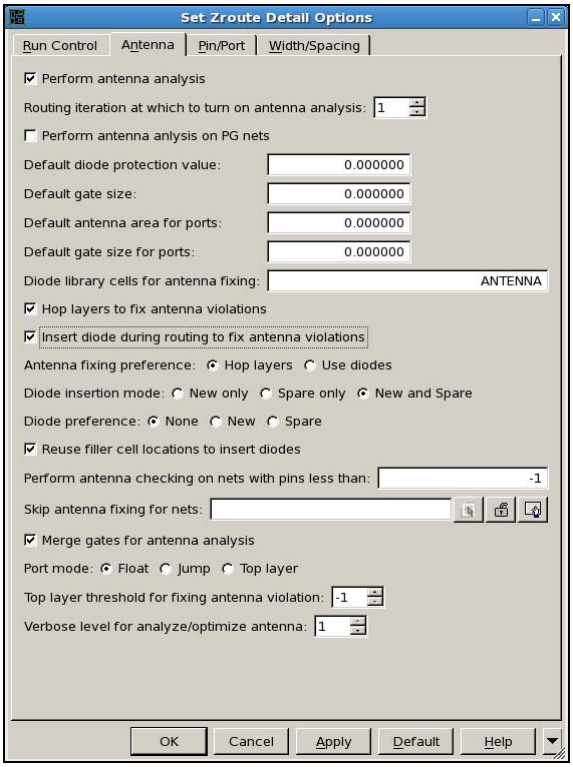
先設定插入 antenna diode 的設定

“ Route > Routing Setup > Set Detail Route Options… ”

選擇 Antenna Tab

Diode library cells for antenna fixing	`ANTENNA`
Insert diode during routing to fix antenna violations	`enable`
Other	`Default value`

按 OK 離開。



再執行 Antenna Violation Fixing

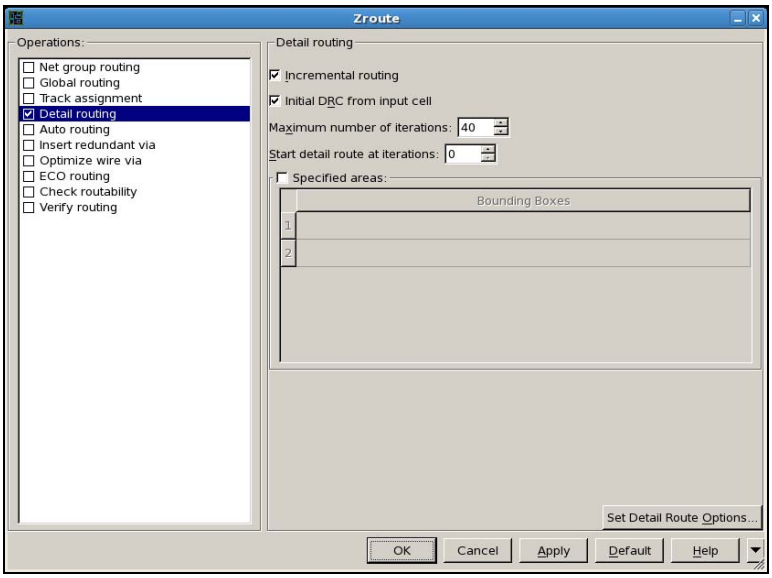
“Route > Verify Route…”

按 OK。

“Route > Detail Route”

Incremental routing	`enable`
Initial DRC from input cell	`enable`

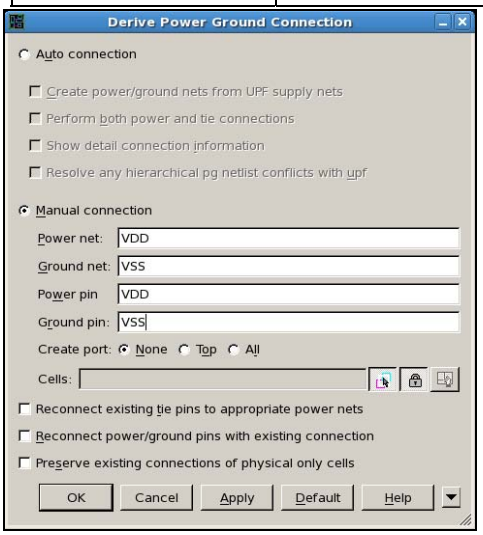
按 OK。



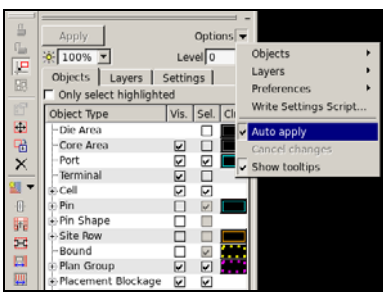
8. Reconnect PG net

“ Preroute > Derive PG Connection ”

Manual connect	`selected`
Power net	VDD
Ground net	VSS
Power pin	VDD
Ground pin	VSS
Other	`Default value`



9. 將 View Settings Options 裡的 Auto apply 打勾

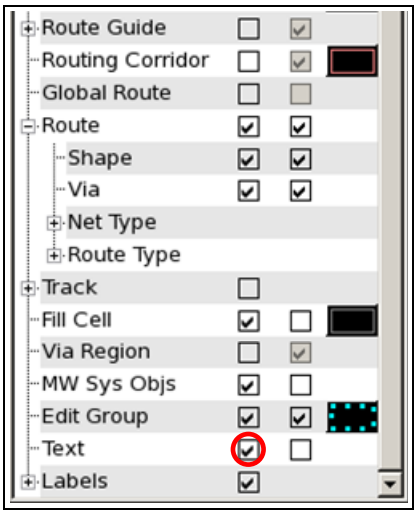


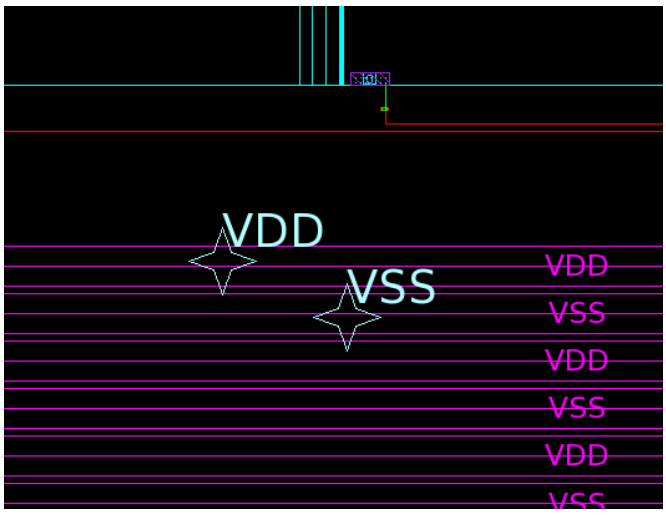
10. Add IO text

在 Message/Input Area 輸入

```
source ../scripts/add_io_text.tcl
add_io_text TEXT3 5 portName
```

將 Core Power 打上 text，必須打在 design 的 Ring 上，VDD、VSS 各打 1 個就好。

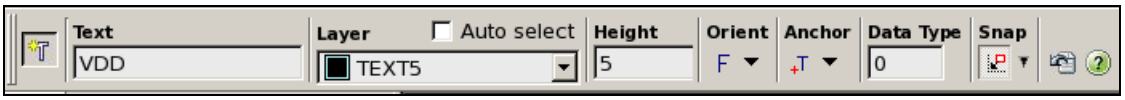




“Edit >Create > Text…” or Shift+t Auto select 請先 disable

Text	VDD
Height	5
Layer	TEXT5

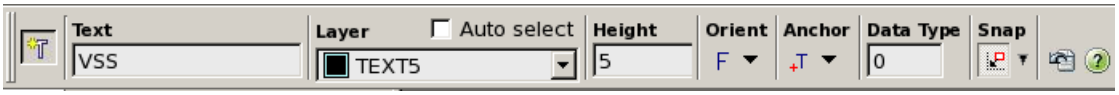
選定要打上的位置點一下按 OK。



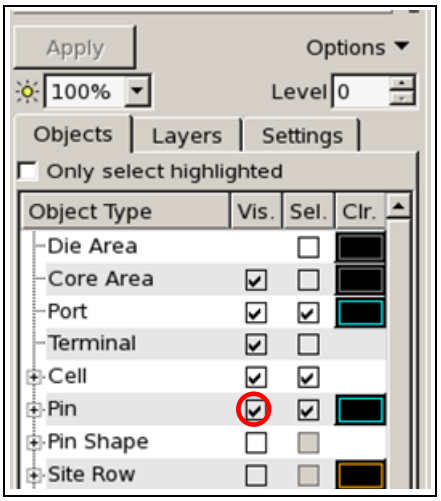
“Edit >Create > Text…” or Shift+t auto select 請先 disable

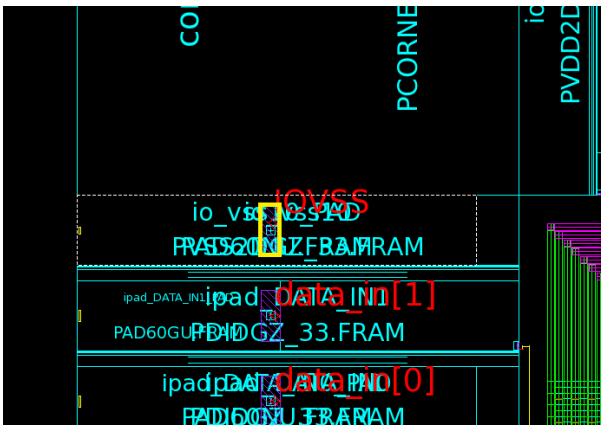
Text	VSS
Height	5
Layer	TEXT5

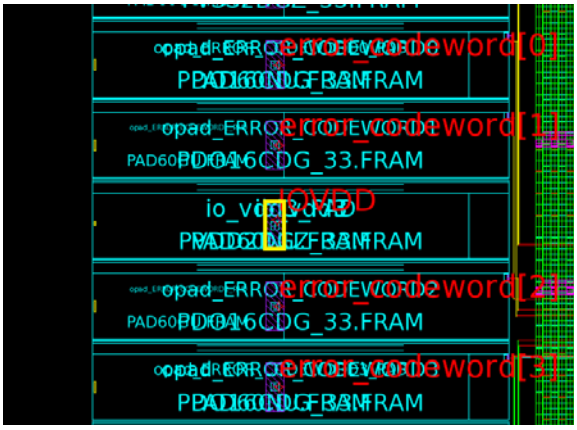
選定要打上的位置點一下按 OK。



將 IO Power 打上 text，必須打在 IO Power Pad 的 true pin 上，IOVDD、IOVSS 各打 1 個就好。



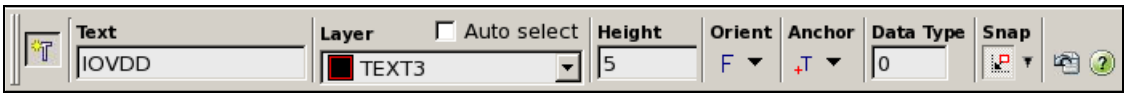




“Edit >Create > Text…” or Shift+t auto select 請先 disable

Text	IOVDD
Height	5
Layer	TEXT3

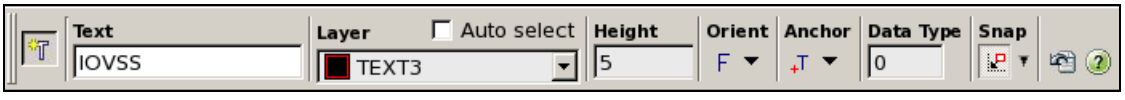
選定要打上的位置點一下按 OK。



“Edit >Create > Text…” or Shift+t auto select 請先 disable

Text	IOVSS
Height	5
Layer	TEXT3

選定要打上的位置點一下按 OK。



11. Add bonding pad

在 Message/Input Area 輸入

```
source ../bond_pads/createNplace_bondpads.tcl
createNplace_bondpads   -inline_pad_ref_name PAD60GU \
                        -stagger true \
                        -stagger_pad_ref_name PAD60NU
```

12. **“ File > Save Design ”**

點選 Save All

“ File > Save Design ”

點選Show advanced options

Save As	Enable
Save As Name	dfm

按 OK。

13. Stream Out GDS II

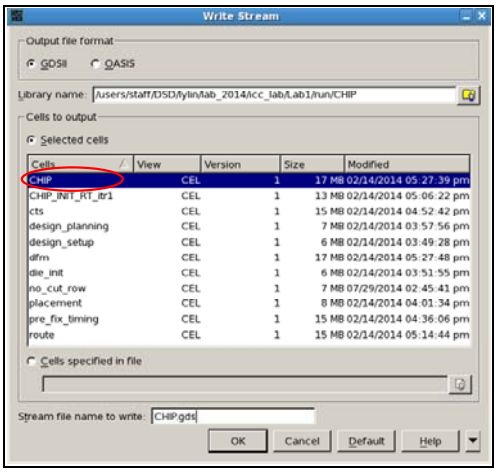
在 Message/Input Area 輸入

```
set_write_stream_options \
-map_layer /cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2/CIC/ICC/macro.map  \
-child_depth 20 -flatten_via
```

“ File > Export > Write Stream…”

Output file format	GDSII
Cells to output	Selected cells “CHIP”
Stream file name to write	CHIP.gds

按 OK。



將 GDS 檔 copy 至 verify 資料夾內

unix% cp ./CHIP.gds ../verify/drc

unix% cp ./CHIP.gds ../verify/lvs

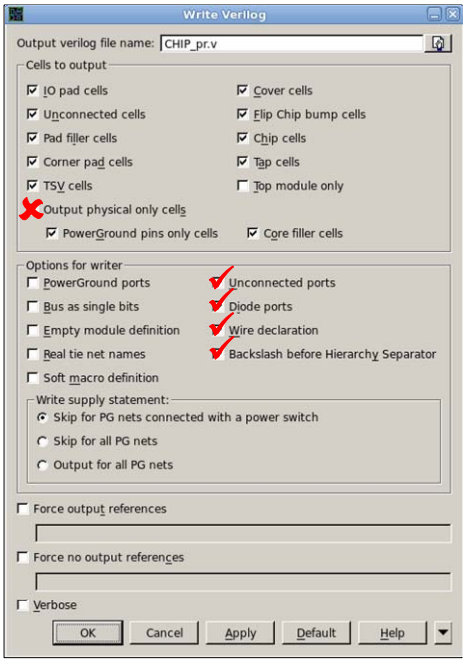
14. Verilog Out for post-layout simulation & LVS

“ File > Export > Write Verilog…”

先按 Default

Output verilog file name	`CHIP_pr.v`
Output physical only cells	`disable`
Unconnected ports	`enable`
Diode ports	`enable`
Wire declaration	`enable`
Backslash before Hierarchy Separator	`enable`

按 OK



將 verilog 檔 copy 至資料夾內

```
unix% cp ./CHIP_pr.v ../verify/lvs
```

15. 輸出 SDF 檔

在 Message/Input Area 輸入

```
write_sdf -version 1.0 -context verilog -load_delay net CHIP_pr.sdf
```

16. **“ File > Save Design**

點選 Save All

17. **“ File > Close Design ”**

“ File > Close Library ”

Lab1-7 Calibre DRC & LVS

1. Change directory to ~/icc_lab/Lab1/verify/drc
   unix% cd ~/icc_lab/Lab1/verify/drc

2. 將 Calibre DRC 的 Rule file copy 至~/icc_lab/Lab1/verify/drc
   unix% cp /cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2/CIC/Calibre/drc/TN90GUTM_DRC.rule .

3. 修改 TN90GUTM_DRC.rule 檔

```
LAYOUT PATH "CHIP.gds"
LAYOUT PRIMARY "CHIP"
```

4. 執行 Calibre DRC
   unix% Qcalibre –drc –hier –turbo_all TN90GUTM_DRC.rule
   當執行”showq”指令觀察 Qentry 執行狀況的 Log 訊息，假若 quser 那行消失，表示 Qcalibre DRC 已經驗證完畢。驗證完畢後會產生” CALIBRE_result”的目錄，可以先觀察 DRC.rep 檔案，觀察 DRC 有無錯誤，應該會看到幾個 DRC error，所有的 density 相關 violation 都可以忽略，可以不予理會這個 DRC error。

   執行 Calibre –rve 分析 DRC 錯誤的數量、原因、坐標等資訊
   由於 Calibre –rve 在 EDA cloud 系統無法直接開啟，必須透過 Laker 軟體間接呼叫 Calibre RVE 工具，步驟如下

5. 將 Qcalibre DRC 自動產生的 CALIBRE_result 目錄內含 DRC_RES.db 檔案，請先複製到自己可以存取的目錄裡。
   例如：unix% cp –r /Queue/Result/14-7-31_xxxxxx_CALIBRE_st301_4082 ~/icc_lab/Lab1/verify/drc

6. 開啟 Laker 軟體
   unix% cd ~/icc_lab/Lab1/verify/drc
   unix% Rlaker

7. **“ File > Import Stream**

Input File Name	CHIP.gds
Top Cell Name	CHIP
Library Name	CHIP
other	default value

   按 OK

   Technology file is not specified. Use default ?
   按 Yes

8. **“ File > Open**

Library	CHIP
Cell	CHIP
View	layout

按 OK

9. 開啟 Calibre RVE
“ Verify > Calibre > Start RVE...

10. 選擇第 5 步驟複製的目錄下的 DRC_RES.db 檔案
按 OK
即可開啟 Qcalibre DRC 驗證結果。

11. Change directory to ~/icc_lab/Lab1/verify/lvs
unix% cd ~/icc_lab/Lab1/verify/lvs

12. 將 Calibre LVS 相關檔案複製到~/icc_lab/Lab1/verify/lvs
unix% cp /cad/CBDK/CBDK_TSMC90GUTM_Arm_v1.2/CIC/Calibre/lvs/* .

13. 產生 RAM Black Box 的 Verilog 和 Spice 檔
修改原來的 nco_table_cos.v 檔，只留下 module、input、output 宣告和 endmodule，如還有其他 Memory 以此類推。

nco_table_cos.v (已編輯完畢)

```
module nco_table_cos (Q, CLK, CEN, A);
    output [15:0]          Q;
    input                  CLK;
    input                  CEN;
    input [9:0]            A;
endmodule
```

利用 v2lvs 產生 nco_table_cos.spi 檔(已產生)
unix% Rv2lvs –v nco_table_cos.v –o nco_table_cos.spi

nco_table_cos.spi

```
$ Spice netlist generated by v2lvs
$ v2012.3_23.18
.SUBCKT nco_table_cos Q[15] Q[14] Q[13] Q[12] Q[11] Q[10] Q[9] Q[8] Q[7] Q[6]
+ Q[5] Q[4] Q[3] Q[2] Q[1] Q[0] CLK CEN A[9] A[8] A[7] A[6] A[5] A[4] A[3] A[2]
+ A[1] A[0]
.ENDS
```

14. 修改 CHIP_pr.v 將所有 PVDD 與 PVSS 關鍵字的 pad 註解

```
.....
//PVSS2DGZ_33 io_vss3 (.VSSPST33 ( SYNOPSYS_UNCONNECTED_156 ) ) ;
//PVSS2DGZ_33 io_vss2 (.VSSPST33 ( SYNOPSYS_UNCONNECTED_157 ) ) ;
//PVSS2DGZ_33 io_vss1 (.VSSPST33 ( SYNOPSYS_UNCONNECTED_158 ) ) ;
//PVDD2DGZ_33 io_vdd11 (.VDDPST33 ( SYNOPSYS_UNCONNECTED_159 ) ) ;
//PVDD2DGZ_33 io_vdd10 (.VDDPST33 ( SYNOPSYS_UNCONNECTED_160 ) ) ;
//PVDD2DGZ_33 io_vdd9 (.VDDPST33 ( SYNOPSYS_UNCONNECTED_161 ) ) ;
.....
```

15. 將 CHIP_pr.v 轉換成 spice 格式

```
Rv2lvs -v CHIP_pr.v -l tpzn90gv3_lvs.v -l tsmc090nvt_fram_lvs.v -l tsmc090hvt_fram_lvs.v -l nco_table_cos.v -l nco_table_log.v -s tpzn90gv3_lvs.spi -s tsmc090nvt_fram_lvs.spi -s tsmc090hvt_fram_lvs.spi -s nco_table_cos.spi -s nco_table_log.spi -o source.spi -s1 VDD -s0 VSS
```

16. 修改 TN90GUTM_LVS_CB.rule

```
LAYOUT PRIMARY "CHIP"
LAYOUT PATH       "CHIP.gds"
LAYOUT CASE YES

SOURCE PRIMARY "CHIP"
SOURCE PATH       "source.spi"
SOURCE CASE YES

VIRTUAL CONNECT BOX NAME "?"
```

移除 IO Power & Ground 相關

```
//LVS BOX    PVDD2DGZ_33
//LVS BOX    PVSS2DGZ_33
//LVS BOX    PVDD2POC_33
```

再移除 CORE Power & Ground 相關

```
//LVS BOX    PVDD1DGZ_33
//LVS BOX    PVSS1DGZ_33
//LVS BOX    PVDD1ANA_33
```

並在最後新增兩行

```
LVS BOX nco_table_log
LVS BOX nco_table_cos
```

17. 執行 Calibre LVS

unix% Qcalibre –lvs –spice layout.spi –hier –turbo_all –auto TN90GUTM_LVS_CB.rule

當執行”showq”指令觀察 Qentry 執行狀況的 Log 訊息，假若 quser 那行消失，表示 Qcalibre LVS 已經驗證完畢。驗證完畢後會產生” CALIBRE_result”的目錄，觀察 lvs.rep 找到 OVERALL COMPARISON RESULT 部份是否 match